Web Feeds Recommending System based on social data

Diego Oliveira Rodrigues[1], Sagar Gurung[1], Dr. Mihaela Cocea[1]

[1]School of Computing – University of Portsmouth (UoP)
Portsmouth – U. K.

{up751567, ece00273}@myport.ac.uk, mihaela.cocea@port.ac.uk

Abstract. *Staying up to date with an individual's interests is a daunting task. An unstoppable number of growing online resources such as websites, blogs and news portals are providing information on various subjects and topics and it is provoking certain difficulties on individuals to identify the best resources or providers in order to trust and follow them continuously. Even though, some trustworthy useful sources have been identified, individuals are still losing them due to lack of management, not bookmarking the sites, carelessness, and so on. The proposed system in this paper will help internet users to figure out an approach to identify and recommend web feeders based on the social data collected from social channels like Facebook. It aims to provide a common place for internet users to read updates of their interest without having to perform search queries. It will also explain about the techniques used to filter data to generate recommended web feeds for users.*

1. Introduction

Due to the expansion of the internet in amount of content and scope [1] it is becoming more and more difficult to identify the right set of information sources to consume and keep updated. It is hard to find out good sources to rely on. This kind of problem is often addressed with information retrieval techniques that allow internet users to provide queries to look for documents containing specific keywords, like in [2] and [3]. However, this kind of approach do not deal entirely with the problem, to create a way to provide to these users personalized recommendations and predictions over large sets of information, the Recommender Systems has come into existence [4].

The Recommender Systems is divided mainly in two categories: Content-based and Collaborative-filtering [5], which are discussed more in details below. There are also some hybrid approaches that use both techniques of these two categories, however, this paper will focus on collaborative filtering to recommend web feeds to internet users. In the paper, social data will be used to analyze the behavior of the users and compare with other users using collaborative-filtering techniques to recommend these web feeds. Before we start discussing about our solution, there is some background knowledge about Recommender Systems presented in the section 2. Afterwards, the section 3 shows the hypotheses that are expected from the proposed solution. This paper focuses a prototype; a used methodology to perform this test is shown on section 4. Then, in section 5 are presented the results collected. Finally are shown the limitations and conclusions of the paper, in sections 6 and 7.

2. Image Segmentation

Recommender Systems is a term that describes any kind of system that "produces individualized recommendations as output, or has the effect of guiding the user in a personalized way to interesting or useful objects in a large space of possible options" [6]. This kind of system emerged as a response to help internet users to observe the best samples in large sets of information available. These large sets of information are presents in many places, such as online stores (i.e. sets of products) or some web services (i.e. shared videos or music). This vast amount of items could make impossible to find out specific information. However, the use of recommender systems creates a way to show up this information to people who are interested in it.

Several techniques are used to perform the recommendations, from heuristics and machine learning to techniques that explores social data. These techniques, when used as recommender systems, are divided in two broad groups: (1) Content-based; and (2) Collaborative-filtering. In Content-based approaches, the users will receive item recommendations similar to the ones their preferred in the past [7]. In another hand, in collaborative recommendations, the users will receive recommendations based on likes of users with similar interests [7]. Also despite these two categories there are hybrid approaches that combine methods of both previous approaches mentioned.

2.1 Content-Based

In Content-based recommendations, the utility of an specific item i to a user u is estimated based on the utilities of a set of other items that are considered similar to i, and have been rated by u [7]. This utility measures the probability of u to become interested in i. An example of this method is given on video sharing platforms (e. g. YouTube, Vimeo). When someone shows that a video is "useful" (i.e. to show utility could mean: given a "like", a share or just watching all the video) the system will begin to show similar videos to others to watch.

The similarities on those videos could be evaluated in several ways. One common way is to use keywords that represent the video. For instance, a video called "Hearthstone card game funny and lucky moments", it is a video from an online card game. By looking at just the title, an algorithm could identify the keywords "Hearthstone" and "Card game" that represents videos. In this way, whether someone will watch the video or not, and gives "like", the system will recommend other similar videos from the Hearthstone game, or maybe from another card games. Other attributes of this video could be used to represent and compare with other movies, such as: its duration, performers, uploaders, etc. Several approaches could also be utilized to evaluate the similarities between the representation of the videos, like the Cosine [8] or Jaccard [9] algorithms.

2.2 Collaborative-Filtering

In contrast to Content-based technique, in collaborative-filtering the utility of an item i, to a user u, is estimated using the rates that others users similar to u assigned to i [7]. Also in video sharing platforms, we have samples of the usage of collaborative-filtering. To use this kind of techniques, first, the system must collect data from the users who will receive the recommendations and build their profiles. Once profiles are created,

they can be compared with others profiles stored within the system, looking for similar users. In this scenario, it is assumed that similar users probably share some of their preferences, which means they probably would like same sort of videos.

For instance, say, user u_0 watched the video, "Hearthstone card game funny and lucky moments" and marked it with "like". Similarly, assuming u_0 have also watched other various videos from Hearthstone card game and "liked" them. Furthermore, there is another user u_s who watched same videos from Hearthstone and gave "likes" to them, and also watched Magic Card Game videos. Since both users (u_0 and u_s) seems to like Hearthstone videos, they could be considered with similar interest by the system. Once they are considered similar, the system could recommend the videos of Magic Card Game ("liked" by u_s) to u_0, because they probably share some preferences. This kind of approach allows recommendations of new kinds of items (e.g. videos), instead of just recommending similar items followed by content-based approach.

3. Hypotheses

The hypothesis will lead us to know if the researched data with the help of the prototype product will be able to: (1) capture data from the social channels, this case - Facebook and mainly focusing on Music band and artists. However, this will be shown as an assumption only; (2) process the collected data according to the structure of an algorithm; (3) process the collected data according to the structure of an algorithm; (4) generate preliminary output to test if it is matching an algorithm; and finally, (5) to generate recommended web feeds to internet users.

If properly filtered recommended web feeds is successfully generated, it can be stored into the RSS database which can be retrieved later to display into the web browser. Once it is being made live, the users should be able to view the list of recommended web feeds on the relevant site and they will click on the links as needed. Moreover, at last but not least, users should be able to rate the web feeds they clicked to check if they really liked it or not to help us find out an effectiveness of recommended system, to cater more into it for better improvements, and so on. However, if they dislikes our recommended web feeds then, we will be reviewing it and try find an alternative solutions i.e. algorithms; if no alternative solutions found then, it will be discarded.

There could be some possible risks to the system as there are always someone out there performing spoofing, shipping, DOS, and many other varieties of attacks in order to hack the system, download malicious rootkits (stealthy/malicious types of software), to steal identities and so on. Due to this, the system will be made as secure as possible to prevent from all those attacks, if further development is required.

4. Methodology

4.1 Process of the proposed system

The proposed system aims to make recommendation of web feeds to internet users. To find out the users preferences, the system will collect data from social channel like Facebook and build up profiles to analyse those preferences, and compares with other preferences from various other users. These data are collected with help of the Facebook

Graph API that enables developers to access the data of their users since these users allow it. This data is extracted from the like pages of the users. From those pages, the system extracts keywords that represent the preferences of the users. Once the system has data, the recommendation algorithm takes place to occur and recommends web feeds to the final users of the proposed system. The Figure 1 shows how it works.

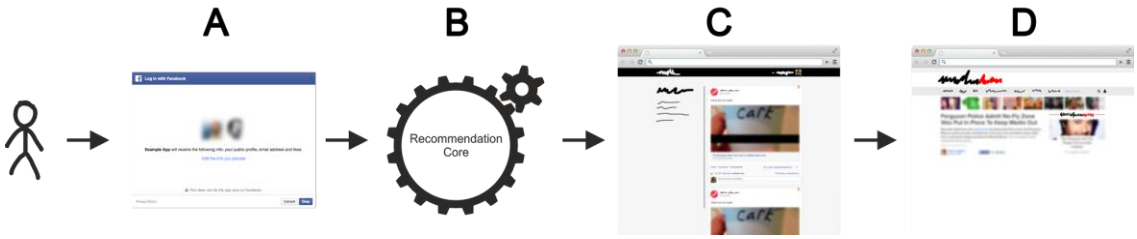

Figure 1 - User flow into the system

The Figure 1 shows the data flow inside the system. First of all, a random user of the proposed system should give access to his information on Facebook through the graph API (A). This user will allow the system to collect the data from his "likes". Since the user allows the system to collect his data, the recommendation algorithm identifies a set of indexed RSS feeds (B) from the web that matches with the users preferences. This recommendation happens using a hybrid approach, combining content-based and collaborative filtering. Once this set of feeds is identified, the results will be displayed to the users in the form of user-interface of the proposed system (C). From user-interface, it is possible to see the summaries of contents of feeds, where users could click on it and takes it to an external site (D) to see the article in details.

To make possible to the users visit web sites related to their preferences, the system was divided in three sections: (1) responsible for collecting and indexing web feeds, (2) responsible for collecting social data from users, and finally, (3) responsible to make recommendations. In this paper, we focused mainly on recommendation section. Figure 2 represents relation between above three sections.

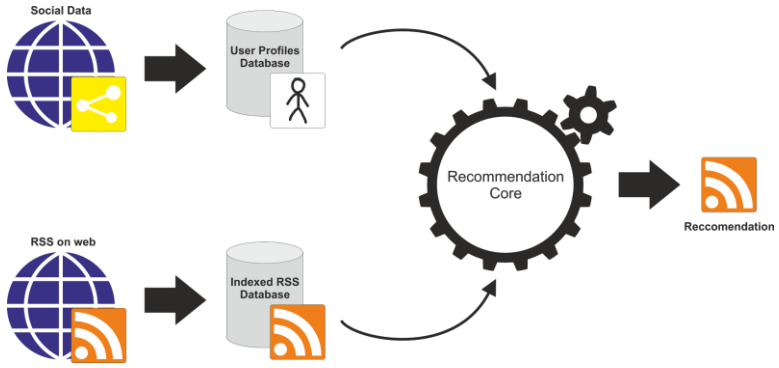

Figure 2 - Proposed system structure

In Figure 2, it represents that the first two sections are responsible for providing data to the recommendation core of the system. This core will analyze the User Profile Database looking for users with similar preferences (i.e. similar set of likes in Facebook). When the system identifies similar preferences, it creates recommendation feeds. How it was mentioned above is that the system uses a hybrid approach to make recommendations. The content-based approach will be used in proposed system to deal

with the cold start problem. However, in this paper the main focus is on the collaborative filtering techniques, which are used to build up the prototype that is presented in the next section.

4.2 Prototyping

To test the proposed use of collaborative filtering techniques in this paper, a prototype algorithm has been built. To perform the test, a particular topic was chosen to elaborate the data, thus, we decided to choose music. Therefore, this paper will use music bands and artists to analyze how an algorithm works. Once the topic was chosen and to perform the test, all it is required is to collect some users' information such as profiles that represents preferences of users about music. Those profiles consist of user identifier and some keywords that describe singers and music bands. Profiles are stored in a JSON format to allow an easy way to read and analyze it.

Since, there is correlation of profiles, the algorithm works by comparing them one by one. The Cosine Algorithm [3] is used to perform and evaluate the similarities between the profiles and the reason for choosing this algorithm is because of its simplicity it provides to build prototypes in order to test proposed approaches, however, in future, alternative well-structured algorithms could be considered to improve the results. Nevertheless, to use this algorithm, the user profiles are represented as vectors, and those vectors are compared between each other.

After the comparison of the profiles, the algorithm looks for the most similar matches. On this matches, the algorithm identifies which of the preferences of the users are not shared. Those not shared preferences are cross-recommended to the users.

5. Results

To test the prototype some user profile data was created. This data are in JSON format containing user identifiers and their preferences. The Table 1 shows a representation of some profiles of users.

John Smith	**David Smith**	**Robert Smith**	**Paul Smith**
Red Hot Chilli Peppers	Franz Ferdinand	The Killers	Imagine Dragons
Guns N Roses	Wolf Gang	Coldplay	Red Hot Chilli Peppers
Lady Gaga	Lana del Rey	The Beatles	Lana del Rey
Imagine Dragons	The Black Keys	Maroon 5	Guns N Roses
Calvin Harris	Gotye	OneRepublic	Maroon 5

Table 1 - Sample profiles of users

The actual sample set consists in a greater number of profiles, each one with more data, than presented in the Table 1. When the algorithm starts to process, it compiles above data which are in JSON format and creates vectors with all the

preferences (i.e. the names of the artists or bands). Once vectors are created, algorithm checks each users' preferences and build new vectors to represent them. At this phase, preferences are dimensions, or indexes, of vectors and it is assigned to 1 if the user had some preferences in his profile and 0 otherwise. The Table 2 shows the main vectors with preferences and two vectors that represents two users in Table 1.

Main Vector	**John Smith**	**Paul Smith**
Red Hot Chilli Peppers	1	1
Guns N Roses	1	1
Lady Gaga	1	0
Imagine Dragons	1	1
Calvin Harris	1	0
Lana del Rey	0	1
...	...	...
Maroon 5	0	0
OneRepublic	0	0

Table 2 - Representation of the main vector with the preferences and some vectors of the profiles of users

Since the algorithm already created these vectors to represent the users it start to compare them two by two using the Cosine similarity approach. In this case, shown in tables above, two users that are more similar are John and Paul. Once they are elected as a match found, preferences of them are cross recommended. The result of this cross recommendation is shown in the Table 3.

John Smith	**Paul Smith**
Maroon 5	Calvin Harris
Lana del Rey	Lady Gaga

Table 3 - Results of the cross recommendation made with the data presented in Table 1

If the proposed system becomes reality, those recommendations will be stored and showed to the users when they login. The users then could rate those recommendations and if they want, they could follow feeds related to them.

6. Limitations

There could be some unforeseen limitations to the solutions, such as: (1) performance of the system issues while collecting data from huge number of users that has been integrated into the social channel, this case, Facebook; (2) errors may occur while running the system due to proxy settings in browsers; (3) initial prototype algorithm

design may not function well as expected due to user preferences changing in daily basis [11]; and (4) as some plugins may be implemented into the system, it may not work as expected [11].

To evaluate the approach, a prototype has been created and also some profiles of users. Those profiles were compared and the algorithm outputs some results. In future papers, the prototype should be analyzed with real data of users. Once with real data, the accuracy of the recommendations could be evaluated because the users that will provide this data could evaluate the recommendations generated by the algorithm.

8. Conclusion

This paper proposes a system that to use collaborative filtering to recommend web feeds to internet users. This proposed system consists in three pieces: one for profile information collection; another to index RSS feeds, and finally one to perform the analysis of the profiles and recommend. The focus here is to present the collaborative filtering approach that is used in this system. This approach creates users profiles from social data, collected from social channels like Facebook, and compare these profiles to find out matches that has similar preferences. Once the algorithm identifies a match, their preferences are cross-recommended allowing the users to know new web feeds they probably would like to follow.

The proposed approach uses information retrieval based techniques to perform the recommendations. The profiles of the users are built from keywords and these techniques are used to evaluate the similarity between them. The Vector Model approach is used to create a representation of profiles that can be evaluated, and then, the Cosine algorithm is used to compare vectors. Several other approaches could be used to perform this analysis, like Jaccard algorithm, Bayesian Networks, and so on. The Cosine algorithm was chosen because it provides a simple way to build the prototype to evaluate the accuracy of the approach proposed in this paper.

References

[1] Silva, E.M., 2009. SWEETS: Sistema de Recomendação Especialistas aplicado a Redes Sociais.

[2] Brin, S. & Page, L., 2012. Reprint of: The anatomy of a large-scale hyper textual web search engine. Computer Networks, 56(18), pp.3825–3833. Available at: http://linkinghub.elsevier.com/retrieve/pii/S1389128612003611.

[3] Abiteboul, S. & Vianu, V., 2000. Queries and computation on the web. Theoretical Computer Science, 239(2), pp.231–255. Available at: http://linkinghub.elsevier.com/retrieve/pii/S0304397599002212.

[4] Konstan, J. a. & Riedl, J., 2012. Recommender systems: from algorithms to user experience. User Modeling and User-Adapted Interaction, 22(1-2), pp.101–123. Available at: http://link.springer.com/10.1007/s11257-011-9112-x [Accessed July 14, 2014]..

[5] Ekstrand, M.D., 2010. Collaborative Filtering Recommender Systems. Foundations and Trends® in Human–Computer Interaction, 4(2), pp.81–173. Available at: http://www.nowpublishers.com/product.aspx?product=HCI&doi=1100000009.

[6] Burke, R., 2002. Hybrid Recommender Systems : Survey and Experiments †. User Modeling and User-Adapted Interaction, 12(4), pp.331 – 370. Available at: http://dl.acm.org/citation.cfm?id=586352.

[7] Adomavicius, G. & Tuzhilin, a., 2005. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering, 17(6), pp.734–749. Available at: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1423975.

[8] Baeza-Yates, R.A. & Ribeiro-Neto, B., 1999. Modern Information Retrieval, Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc.

[9] Niwattanakul, S. et al., 2013. Using of Jaccard Coefficient for Keywords Similarity. In Proceedings of the International MultiConference of Engineers and Computer Scientists 2013.

[10] Valerie Coffman, May 21, 2014. [Online] Available from: http://datacommunitydc.org/blog/2013/05/recommendation-engines-why-you-shouldnt-build-one/

[11] Richard Macmanus, Jan 28, 2009. [Online] Available fom: http://readwrite.com/2009/01/28/5_problems_of_recommender_systems